Multistage Shifted JACOBI Spectral Method for solving linear & nonlinear Fractional Differential Equations

Kyung Duk Park

Department of Mathematical Sciences
Graduate School of UNIST

ULSAN NATIONAL INSTITUTE OF SCIENCE AND TECHNOLOGY

Multistage Shifted JACOBI Spectral Method for solving linear & nonlinear Fractional Differential Equations

A thesis
submitted to the Graduate School of UNIST
in partial fulfillment of the
requirements for the degree of
Master of Science

Kyung Duk Park

01.14.2015
Approved by

Major Advisor
Bongsoo Jang

Multistage Shifted JACOBI Spectral Method for solving linear & nonlinear Fractional Differential Equations

Kyung Duk Park

This certifies that the thesis of Kyung Duk Park is approved.

01.14.2015

Thesis Supervisor: Bongsoo Jang

Pilwon Kim: Thesis Committee Member #1

Byoung Ki Seo

Byoung Ki Seo: Thesis Committee Member #2

Abstract

In this work, we developed a new numerical method based on the shifted jacobi polynomials for solving linear and nonlinear initial value problem and boundary value problem of fractional differential equation. We extend the conventional spectral approaches such as the Shifted Jacobi Tau(SJT) method for linear problem and the Shifted Jacobi Collocation(SJC) method for nonlinear problem, by using the multistage methodology. These methods are called the Multistage Shifted Jacobi Tau(M-SJT) and the Multistage Shifted Jacobi Collocation(M-SJC) method,respectively. From the several illustrative examples, the advantages of using the proposed methods are discussed for the initial value problem and we compare the proposed methods with exact solution and conventional spectral approaches.

In addition, we extend the proposed methods for solving nonlinear boundary value problem of the fractional differential equations. Since all proposed methods are developed for solving the linitial problem, it is necessary to convert the boundary problem to the initial problem. Here we adopt the nonlinear shooting method combined with M-SJC. From the numerical example, the advantages of using the proposed methods are discussed for the nonlinear boundary value problem and we compare the proposed methods with exact solution and conventional spectral approaches.

Contents

List of Figures

List of Tables

I

Introduction

The topic of fractional calculus has been investigated as a pure mathematical theory. It was not used in application fields such as physics. One reason could be that there are several definitions of fractional derivatives which are not equivalent. However, during the last decade fractional calculus has been considered as an important mathematical modeling in various fields of applied sciences and engineering. Several numerical methods such as Adomian decomposition method[8], homotopy analysis method[11],[12], homotopy-perturbation method [6, 7], variational iteration method[10] have been introduced to solve fractional differential equations . In this work, we concentrate on the shifted jacobi polynomials for solving the linear and nonlinear fractional problems. For linear initial fractional problems, the Shifted Jacobi Tau (SJT) method are employed to obtain accurate approximations. In SJT method, assuming the solution is spanned by the finite many shifted jacobi polynomials, the approximations are calculated by solving the system which is given by inner product. However, there are some difficulties in obtaining reliable approximation due to the regulaity of the exact solution and the computational inefficiency by using higher order shifted jacobi polynomials. In order to avoid such difficulties we propose the multistage SJT(M-SJT) method. The basic idea of M-SJT method is to apply the standard SJT method to the problem in each sub-domain. Since the fractional derivative is defined by the integral operator, it has not a local operator. In other words, the fractional derivative is a global operator. Thus, it has to be considered to construct a new system that is generated by using inner product in the M-SJT method. Here, we propose the new system in obtaining an approximate solution.

For nonlinear fractional problems it is impossible to construct the system in SJP method. Thus, for the given nonlinear problem, the Shifted Jacobi Collocation (SJC) method has been introduced to find an accurate approximation. In SJC method, assuming the solution is a linear combination of finite many jacobi polynomials, the system that is generated by choosing approriate collocation points is directly solved by the newton's method. Even if the SJC method performs well in obtaining a reliable approximation, it is very difficult to use the newton's method if the higher order degree of jacobi polynomials are employed. In the similar manner to the M-SJT, we propose the multistage approach to find an accurate approximation, so called the Mutistage Shifted Jacobi Collocation (M-SJC) method. In each sub-domain the standard SJC method is applied with low order degree of jacobi polynomials so that it is easy to utilize the newton's method. It is also required to construct a new algorithm in M-SJC method because of the global propertie of the fractional derivative.

We also apply the methods based on the shifted jacobi polynomials for solving fractional boundary value problems. The standard SJC method is applied to obtain an accurate approximation. To enhance the numerical performance we utilize the multistage approach. As seen before, the algorithm of M-SJC method is established on solving the initial value problem of fractional differential equation. To apply the M-SJC method to the boundary value problem, the nonlinear shooting method is introduced. In the nonlinear shooting method, the given boundary value problem is considered to the initial value problem with unknown initial condition that is corrected by solving an auxiliary initial value problem. Combining the M-SJC method with the nonlinear shooting method we illustrated several numerical experiments which show that the proposed method is reliable and efficient for solving boundary value problem of fractional order differential equation.

The paper is organized as follows. In Chapter 2 we present some mathematical preliminaries about fractional calculus and shifted Jacobi polynomial. In Chapter 3, we concentrate on methods which is based on shifted Jacobi polynomials for solving initial value problems of fractional order differential equation such as Shifted Jacobi Collocation method and Shifted Jacobi Tau method. To enhance numerical accuracy the multistage approach is demonstrated. In Chapter 4 we present the Shifted Jacobi Collocation for solving boundary value problem of fractional order differential equation. Also we describe how to apply the multistage approach to the boundary value problems. Finally, we give a conclusion in Chapter 5.

II

Preliminary

In this chapter, we descibe the basic definition and several properties of fractional calculus [4]. Since the main tool for solving initial and boundary value problems of fractional differential equation is based on the shifted jacobi polynomials, their definitions as well as several properites are presented.

2.1 Fractional Calculus

We are to give a first definition for fractional integral and differential operators J_a^n and D^n, $n \notin \mathbb{N}$. As indicated above, we begin with the integral operator [4].

Definition 2.1.1. *Let $n \in \mathbb{R}_+$. The operator J_a^n, defined on $L_1[a,b]$ by*

$$J_a^n f(x) = \frac{1}{\Gamma(n)} \int_a^x (x-t)^{n-1} f(t)\ dt \tag{2.1.1}$$

for $a \leq x \leq b$, is called the Riemann-Liouville fractional integral operator *of order n.*
For $n = 0$, we set $J_a^0 := I$, the identity operator.

Now important properties of integral operator is introduced by Theorem[4].

Theorem 2.1.1. *Let $m, n \geq 0$ and $\phi \in L_1[a,b]$. Then,*

$$J_a^m J_a^n \phi = J_a^{m+n} \phi \tag{2.1.2}$$

holds almost everywhere on $[a,b]$. if additionally $\phi \in C[a,b]$ or $m+n \geq 1$, then the identity holds everywhere on $[a,b]$.

Corollary 2.1.1. *Under the assumptions of theorem (2.1.1)*

$$J_a^m J_a^n \phi = J_a^n J_a^m \phi \tag{2.1.3}$$

There is an algebraic way to state this result.

Having established these fundamental properties of Riemann-Liouville integral operator, we now come to the corresponding differential operators[4].

Definition 2.1.2. *Let $n \in \mathbb{R}_+$ and $m = \lceil n \rceil$. The operator D_a^n, defined by*

$$D_a^n f = D^m J_a^{m-n} f \tag{2.1.4}$$

is called the Riemann-Liouville fractional differential operator *of order n. For $n = 0$, we set $D_a^0 := I$,the identity operator.*

It turns out that the Riemann-Liouville derivatives have certain disavantages when trying to model real-world phenomena with fractional differential equations. We shall therefore now discuss a modified concept of a fractional derivative[4].

Definition 2.1.3. *Assume that $n \geq 0$ and that f is such that $D_a^n[f - T_{m-1}[f;a]]$ exists, where $T_{m-1}[f;a]$ denotes the Taylor polynomial of degree $m-1$ for the function f, centered at a and $m = \lceil n \rceil$. Then we define the function $D_{*a}^n f$ by*

$$D_{*a}^n f = D_a^n[f - T_{m-1}[f;a]]. \tag{2.1.5}$$

The operator D_{*a}^n is called the Caputo differential operator *of order n.*

Actually this concept has been introduced independently by many authors.

Lemma 2.1.1. *If $m-1 < n \leqslant m,\ m \in \mathbb{N}$, then,*

$$1.R - L : D^n J^n f(x) = f(x) \tag{2.1.6}$$

$$2.Caputo : J^n D^n f(x) = f(x) - \sum_{k=0}^{m-1} \frac{d^k}{dx^k}(0^+)\frac{t-a}{k!}. \tag{2.1.7}$$

Lemma(2.1.1) is the differences between Riemann-Liouville and Caputo. Caputo fractional differential operator has a initial condition. So we will try to model real-world phenomena with fractional differential equations.

We need to present that functions will turn out to be of fundamental importance. We begin with the more restrictive of the two concepts.

Definition 2.1.4. *Let $n > 0$. The function E_n defined by*

$$E_n(z) := \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(jn+1)}, \tag{2.1.8}$$

whenever the series converges is called the Mittag-Leffler function *of order n.*

This function has been introduced by *Mittag-Leffler.* We immediately notice that

$$E_1(z) = \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(j+1)} = \sum_{j=0}^{\infty} \frac{z^j}{j!} = \exp(z)$$

is just the well known exponential function.
The more general class of functions is defined as follows.

Definition 2.1.5. *Let $n_1, n_2 > 0$. The function E_{n_1,n_2} defined by*

$$E_{n_1,n_2}(z) := \sum_{j=0}^{\infty} \frac{z^j}{\Gamma(jn_1 + n_2)}, \tag{2.1.9}$$

whenever the series converges is called the two-parameter Mittag-Leffler function *with parameters n_1 and n_2.*

It essentially sates that the eigenfunction of Caputo differential operators may be expressed in terms of Mittag-Leffler functions.

Theorem 2.1.2. *Let $n > 0$ and $\lambda \in \mathbb{R}$. Moreover define*

$$y(x) := E_n(\lambda x^n), \qquad x \geq 0. \tag{2.1.10}$$

Then,

$$D^n_{*a} y(x) = \lambda y(x). \tag{2.1.11}$$

Proof. We first look at the case $\lambda = 0$ and note that in this case $y(x) = E_n(0) = 1$. Hence, $D^n_{*0} y(x) = 0 = \lambda y(x)$ as required If, on the other hand, $\lambda \neq 0$, then(using the notation

$p_k(x) := x^k$)

$$\begin{aligned} D^n_{*0}y(x) &= D^n_{*0}\Big[\sum_{j=0}^{\infty}\frac{(\lambda p_n)^j}{\Gamma(1+jn)}\Big](x) = J_0^{m-n}D^m\Big[\sum_{j=0}^{\infty}\frac{\lambda^j p_{nj}}{\Gamma(1+jn)}\Big](x) \\ &= J_0^{m-n}\Big[\sum_{j=0}^{\infty}\frac{\lambda^j D^m p_{nj}}{\Gamma(1+jn)}\Big](x) = J_0^{m-n}\Big[\sum_{j=1}^{\infty}\frac{\lambda^j D^m p_{nj}}{\Gamma(1+jn)}\Big](x) \\ &= J_0^{m-n}\Big[\sum_{j=1}^{\infty}\frac{\lambda^j p_{nj-m}}{\Gamma(1+jn-m)}\Big](x) = \sum_{j=1}^{\infty}\frac{\lambda^j J_0^{m-n}p_{nj-m}(x)}{\Gamma(1+jn-m)} \\ &= \sum_{j=1}^{\infty}\frac{\lambda^j p_{nj-n}(x)}{\Gamma(1+jn-n)} = \sum_{j=1}^{\infty}\frac{\lambda^j x^{nj-n}}{\Gamma(1+jn-n)} \\ &= \sum_{j=0}^{\infty}\frac{\lambda^{j+1}x^{nj}}{\Gamma(1+jn)} = \lambda y(x). \end{aligned} \tag{2.1.12}$$

Here, we have used the fact that, in view of the convergence properties of the series defining the Mittage-Leffler function, we may interchange first summation and differentiation and later summation and integration. □

It is frequently of interest to have some knowledge about the asymptotic behaviour of the Mittag-Leffler functions.

Theorem 2.1.3. *Let $n > 0$. The mittag-Leffler function E_n behaves as follows:*
(a) $E_n(re^{i\phi}) \to 0$ for $r \to \infty$ if $\mid \phi \mid > n\pi/2$.
(b) $E_n(re^{i\phi})$ remains bounded for $r \to \infty$ if $\mid \phi \mid = n\pi/2$.
(c) $\mid E_n(re^{i\phi}) \mid \to \infty$ for $r \to \infty$ if $\mid \phi \mid < n\pi/2$.
Obviously, in the classical case $n = 1$ this reduces to the well known fact that, as $\mid z \mid \to \infty$, $exp(z)$ (a) goes to zero if $arg\ z > \pi/2$, (b) remains bounded if $argz = \Pi/2$ and (c)grows without bound if $argz < \Pi/2$.

We consider some theorems about Mittag-Leffler functions.

Theorem 2.1.4. *Let $n \in \mathbb{R}$, $m-1 < n < m$, $m \in \mathbb{N}$, $\lambda \in \mathbb{C}$, then the Caputo fractional derivative of the exponential function has the form*

$$D^n_* e^{\lambda t} = \sum_{k=0}^{\infty}\frac{\lambda^{k+m}t^{k+m-n}}{\Gamma(k+1_m-n)} = \lambda^m t^{m-n}E_{1,m-n+1}(\lambda t), \tag{2.1.13}$$

where $E_{a,b}(z)$ is the two-parameter function of Mittag-Leffler type.

Theorem 2.1.5. *Let $n \in \mathbb{R}$, $m-1 < n < m$, $m \in \mathbb{N}$, $\lambda \in \mathbb{C}$, then*

$$D^n_* \sin(\lambda t) = -\frac{i(i\lambda)^m t^{m-n}}{2}(E_{1,m-n+1}(i\lambda t) - (-1)^m E_{1,m-n+1}(-i\lambda t)). \tag{2.1.14}$$

Proof. The following representation of the sine function is used

$$\sin z = \frac{e^{iz} - e^{-iz}}{2i}, \qquad z \in \mathbb{C}.$$

Then,

$$\begin{aligned} D_*^n \sin(\lambda t) &= D_*^n \frac{e^{iz} - e^{-iz}}{2i} \\ &= \frac{1}{2i}(D_*^n e^{i\lambda t} - D_*^n e^{-i\lambda t}) \\ &= -\frac{i(i\lambda)^m t^{m-n}}{2}(E_{1,m-n+1}(i\lambda t) - (-1)^m E_{1,m-n+1}(-i\lambda t)). \end{aligned}$$

□

2.2 Shifted Jacobi Polynomials

The well-known Jacobi polynomials associated with the parameters are a sequence of polynomials $P_n^{(\alpha,\beta)}(t)(n = 0, 1, 2, \dots)$,each respectively of degree n, are defined by

$$P_n^{(\alpha,\beta)}(t) = \sum_{m=0}^{n} \frac{\Gamma(\alpha+n+1)\Gamma(\alpha+\beta+n+m)}{\Gamma(\alpha+\beta+n+1)\Gamma(\alpha+m+1)m!(n-m)!}(\frac{t-1}{2})^m \quad \text{in}(-1,1).$$

Then, it is every to see that $P_n^{(\alpha,\beta)}(-1) = (-1)^n \frac{\Gamma(n+\beta+1)}{\Gamma(\beta+1)n!}$, $P_n^{(\alpha,\beta)}(1) = \frac{\Gamma(n+\alpha+1)}{\Gamma(\alpha+1)n!}$. For using Jacobi polynomials on $\Omega := (a, b)$, we present the shifted Jacobi polynomials by implementing the change of variable $t = (\frac{2(x-a)}{b-a} - 1)$. Let the shifted Jacobi polynomials $P_i^{(\alpha,\beta)}(\frac{2(x-a)}{b-a} - 1)$ be denoted by $P_{\Omega,i}^{(\alpha,\beta)}(x)$, satisfying the orthogonality relation

$$\int_a^b w_\Omega^{(\alpha,\beta)} P_{\Omega,j}^{(\alpha,\beta)} P_{\Omega,i}^{(\alpha,\beta)} = h_j, \tag{2.2.1}$$

where $w_\Omega^{(\alpha,\beta)}(x) = \left(2 - \frac{2(x-a)}{(b-a)}\right)^\alpha \left(\frac{2(x-a)}{(b-a)}\right)^\beta$. Here, h_j is defined by $h_j = \begin{cases} c_j, & \text{if } i = j, \\ 0, & \text{if } i \neq j, \end{cases}$ where c_j are constants. Then, the shifted Jacobi polynomial $P_{\Omega,n}^{(\alpha,\beta)}(x)$ of degree n has the form

$$P_{\Omega,n}^{(\alpha,\beta)}(x) = \sum_{k=0}^{n} \frac{\Gamma(n+\alpha+1)\Gamma(n+k+\alpha+\beta+1)}{\Gamma(k+\alpha+1)\Gamma(n+\alpha+\beta+1)(n-k)!k!(b-a)^k}(x-b)^k, \tag{2.2.2}$$

where $P_{\Omega,n}^{(\alpha,\beta)}(a) = (-1)^n \frac{\Gamma(n+\beta+1)}{\Gamma(\beta+1)n!}$, $P_{\Omega,n}^{(\alpha,\beta)}(b) = \frac{\Gamma(n+\alpha+1)}{\Gamma(\alpha+1)n!}$. We notice that for the square integrable function $u_\Omega(x)$ in $\Omega = (a, b)$ can be expanded in terms of shifted Jacobi polynomials as

$$u_\Omega(x) = \sum_{i=0}^{\infty} \xi_i P_{\Omega,i}^{(\alpha,\beta)}(x).$$

Then, the coefficients ξ_i can be determined by

$$\xi_i = \frac{1}{h_i} \int_a^b w_\Omega^{\alpha,\beta}(x) u_\Omega(x) P_{\Omega,i}^{(\alpha,\beta)}(x) dx \qquad i = 0, 1, \cdots .$$

Now suppose that the function $u_\Omega(x)$ can be approximated by using $(N+1)$ terms of shifted Jacobi polynomials.

$$u_{\Omega(x),N} = \sum_{i=0}^{N} \xi_i P_{\Omega,i}^{\alpha,\beta}(x). \tag{2.2.3}$$

Among these Shifted Jacobi polynomials, the most commonly used Shifted Jacobi polynomials in numerical analysis are the shifted Legendre polynomials $P_{\Omega,n}$; and the shifted Chebyshev polynomials of the first kind $T_{\Omega,n}$.

$$P_{\Omega,n}(x) = P_{\Omega,n}^{(0,0)}(x), T_{\Omega,n}(x) = \frac{n!\Gamma(1/2)}{\Gamma(n+1/2)} P_{\Omega,n}^{(-\frac{1}{2},-\frac{1}{2})}(x).$$

Moreover, the Shifted Chebyshev polynomials of the first kind $T_{\Omega,n}$ can be written on

$$T_{\Omega,n}(x) =_2 F_1(-n, n; \frac{1}{2}; \frac{1}{2}(2 - 2\frac{x-a}{b-a})),$$

where ${}_2F_1(q, b; c; z)$ is Hypergeometric function and $\sum_{n=0}^{\infty} \frac{(q)_n (b)_n}{z}^n (c)_n n!$. Here $(q)_n$ is the Pochhammer symbol, which is defined by

$$(q)_n = \begin{cases} 1, & \text{if } n = 0, \\ q(q+1)\cdots(q+n-1), & \text{if } n > 0. \end{cases}$$

III

Initial Value Problem of Fractional Differential Equation

3.1 Model Problem

Let us consider the linear initial value problem of fractional differential equation as follows

$$\begin{aligned} D_0^\nu u(x) &= f(x), \quad \text{in } \Omega = (0,1) \\ u^{(i)}(0) &= d_i, \quad i = 0, ..., m-1, \end{aligned} \tag{3.1.1}$$

where $m-1 < \nu \leqslant m$ and $m \in \mathbb{N}$. Here, the fractional derivative $D^\nu u(x)$ denotes the Caputo fractional derivative of order ν for $u(x)$, and the values of $d_i (i = 0, \cdots, m-1)$ describe the initial data of $u(x)$ and $f(x)$ is a given source function. We apply the shifted jacobi polynomials to the model (3.1.1) in obtaining an accurate approximation.

3.1.1 Shifted Jacobi Tau (SJT) method for initial value problem

First of all, we define the weighed Sobolev space $L^2_{w_\Omega^{(\alpha,\beta)}}(\Omega)$ by

$$L^2_{w_\Omega^{(\alpha,\beta)}}(\Omega) = \Big\{ f : \int_\Omega w_\Omega^{(\alpha,\beta)}(x) f^2(x) dx < +\infty \Big\}. \tag{3.1.2}$$

We recall that $\{P_{\Omega,j}^{(\alpha,\beta)}(x) : 0 \le j \le N\}$ forms a complete orthogonal system in $L^2_{w_\Omega^{(\alpha,\beta)}}$. Therefore, we set

$$S_N(\Omega) = \text{span}\{1, P_{\Omega,1}^{(\alpha,\beta)}(x), P_{\Omega,2}^{(\alpha,\beta)}(x), \ldots, P_{\Omega,N}^{(\alpha,\beta)}(x)\}.$$

Multiplying both sides of (3.1.1) by the shifted Jacobi polynomial $P_{\Omega,k}^{(\alpha,\beta)}$ and taking the inner product, we have to find $u_N \in S_N(\Omega)$ such that

$$\begin{aligned}&\left(D_0^{\nu}u_N, P_{\Omega,k}^{(\alpha,\beta)}\right)_{w_\Omega^{(\alpha,\beta)}} = \left(f, P_{\Omega,k}^{(\alpha,\beta)}\right)_{w_\Omega^{(\alpha,\beta)}}, \qquad k = 0,1,\ldots,N-m\\ &u_N^{(i)}(0) = d_i, \quad i = 0,1,\ldots,m-1,\end{aligned} \tag{3.1.3}$$

where $w_\Omega^{(\alpha,\beta)} = (2-2x)^\alpha(2x)^\beta$ and $(u,v)_{w_\Omega^{(\alpha,\beta)}} = \int_\Omega w_\Omega^{(\alpha,\beta)}(x)u(x)v(x)dx$ denote the inner product in $L^2_{w_\Omega^{(\alpha,\beta)}}(\Omega)$. Suposse that

$$u_N(x) = \sum_{j=0}^{N} \xi_j P_{\Omega,j}^{(\alpha,\beta)}. \tag{3.1.4}$$

Then, the variational formulation of (3.1.3) is equivalent to

$$\begin{aligned}&\sum_{j=0}^{N} \xi_j (D_0^{\nu} P_{\Omega,j}^{(\alpha,\beta)}, P_{\Omega,k}^{(\alpha,\beta)})_{w_\Omega^{(\alpha,\beta)}} = (f(x), P_{\Omega,k}^{(\alpha,\beta)})_{w_\Omega^{(\alpha,\beta)}},\\ &k = 0,1,\ldots,N-m\end{aligned} \tag{3.1.5}$$

and initial conditions

$$\begin{aligned}&\sum_{j=0}^{N} \xi_j D^{k-N+m-1} P_{\Omega,j}^{(\alpha,\beta)}(0) = d_{k-N+m-1},\\ &k = N-m+1, N-m+2, \ldots, N.\end{aligned} \tag{3.1.6}$$

Denoting that

$$\begin{aligned}&\mathbf{K} = (\xi_0, \xi_1, \ldots, \xi_N)^T\\ &\mathbf{A} = (a_{kj})_{0\le k,j\le N}\\ &\mathbf{F} = (f_0, f_1, \ldots, f_{N-m}, d_0, \ldots, d_{m-1})^T,\end{aligned} \tag{3.1.7}$$

where $a_{kj} = (D_0^{\nu} P_{\Omega,j}^{(\alpha,\beta)}, P_{\Omega,k}^{(\alpha,\beta)})_{w_\Omega^{(\alpha,\beta)}}$ $(0 \le k \le N-m,\ 0 \le j \le N)$, $a_{kj} = D^{k-N+m-1}P_{\Omega,j}^{(\alpha,\beta)}(0)$ $(N-m+1 \le k \le N,\ 0 \le j \le N)$, $f_k = (f(x), P_{\Omega,k}^{(\alpha,\beta)})_{w_\Omega^{(\alpha,\beta)}}$, $(k = 0,1,\ldots,N-m)$. We investigate that (3.1.5) (3.1.6) are equivalent to the matrix system

$$\mathbf{AK} = \mathbf{F}. \tag{3.1.8}$$

3.1.2 Multistage Shifted Jacobi Tau (M-SJT) method for initial value problem

In the previous section we described the basic idea of the SJT method. In the SJT method, it is the key to solve the matrix system (3.1.8). Also it is cleat that the matrix system become larger if the solution is approximated with many shifted jacobi polynomial. That is, N is large in (3.1.8). In this section, we propose an efficient computational method, namely Multistage Shifted Jacobi Tau(M-SJT) method. The basic idea of the M-SJT method is to apply the standard SJT method to the problem in each sub-domain. To describe the method, we consider the equally spaced partition $P : 0 = x_0 < x_1 < \cdots < x_n = 1$, where the nodes $x_l = l * h, h = 1/n$, $l = 0, \cdots, n$. Let us define the lth sub-domain $\Omega_l \equiv (x_{l-1}, x_l)$ and $u(x) \mid_{(\Omega_l)} \equiv u_l(x)$. We recall that $\{P^{(\alpha,\beta)}_{\Omega_l,j}(x) : 0 \leq j \leq N\}$ forms a complete orthogonal system in $L^2_{W^{(\alpha,\beta)}_{\Omega_l}}$,where

$$P^{(\alpha,\beta)}_{\Omega_l,n}(x) = \begin{cases} \sum_{k=0}^{i} \frac{\Gamma(i+\alpha+1)\Gamma(i+k+\alpha+\beta+1)}{\Gamma(k+\alpha+1)\Gamma(i+\alpha+\beta+1)(i-k)!k!(t_l-t_{l-1})^k}(x-t_l)^k, & \text{if } x \in \Omega_l, \\ 0, & \text{if } x \notin \Omega_l. \end{cases}$$

Suppose that all solutions $u_i(x), i = 1, \cdots, l-1$, are approximated by $u_{i,N}(x)$, where $u_{i,N}(x) = \sum_{j=0}^{N} \xi^i_j P^{(\alpha,\beta)}_{\Omega_i,j}(x)$. Now, let us define

$$S_N(\Omega_l) = \text{span}\{1, P^{(\alpha,\beta)}_{\Omega_l,1}(x), P^{(\alpha,\beta)}_{\Omega_l,2}(x), \ldots, P^{(\alpha,\beta)}_{\Omega_l,N}(x)\}.$$

and suppose that an approximation $u_{l,N}(x) \in S_N(\Omega_l)$ of $u_l(x)$ is defined by

$$u_{l,N}(x) = \begin{cases} \sum_{i=0}^{N} \xi^l_i P^{(\alpha,\beta)}_{\Omega_l,i}(x), & \text{if } x \in \Omega_l, \\ 0, & \text{if } x \notin \Omega_l \end{cases}.$$

For $x \in \Omega_l$, we have

$$\begin{aligned} D^\nu_0 u(x) &= \int_{x_0}^{x} (x-s)^{m-\nu-1} D^m u(s) ds \\ &= \sum_{k=0}^{l-1} \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_k(s) ds + \int_{x_{l-1}}^{x} (x-s)^{m-\nu-1} D^m u_l(s) ds \\ &\approx \sum_{k=0}^{l-1} \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_{k,N}(s) ds + \int_{x_{l-1}}^{x} (x-s)^{m-\nu-1} D^m u_{l,N}(s) ds. \end{aligned} \tag{3.1.9}$$

Let us define

$$D^\nu_{\Omega_k} u_{k,N}(x) = \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_{k,N}(s) ds. \tag{3.1.10}$$

Since

$$D^\nu_0 u(x) = f(x), \quad \text{in } (x_0, x_l), \tag{3.1.11}$$

we have for $x \in (x_{l-1}, x_l)$,

$$D^{\nu}_{\Omega_l} u_l(x) = f(x) - \sum_{k=0}^{l-1} D^{\nu}_{\Omega_k} u_k(x). \tag{3.1.12}$$

The standard SJT method to (3.1.12) is to obtain $u_{l,N}(x)$ which satisfies the following

$$\begin{aligned}&\sum_{j=0}^{N} \xi_j^l \Big(D^{\nu}_{t_{l-1}} P^{(\alpha,\beta)}_{\Omega_l,j}(x), P^{(\alpha,\beta)}_{\Omega_l,k}(x) \Big)_{W^{(\alpha,\beta)}_{\Omega_l}} = \Big(g(x), P^{(\alpha,\beta)}_{\Omega_l,k}(x) \Big)_{W^{(\alpha,\beta)}_{\Omega_l}}\\ &k = 0, 1, \ldots, N-m\end{aligned} \tag{3.1.13}$$

and initial conditions

$$\begin{aligned}&\sum_{j=0}^{N} \xi_j D^{k-N+m-1} P^{(\alpha,\beta)}_{\Omega_l,j}(t_{l-1}) = D^{k-N+m-1} u_{l-1,N}(t_{l-1}),\\ &k = N-m+1, N-m+2, \ldots, N\end{aligned} \tag{3.1.14}$$

where

$$g(x) = f(x) - \sum_{k=0}^{l-1} D^{\nu}_{\Omega_k} u_{k,N}(x).$$

Denoting that

$$\begin{aligned}\mathbf{K_l} &= (\xi_0^l, \xi_1^l, \ldots, \xi_N^l)^T\\ \mathbf{A_l} &= (a^l_{kj})_{0 \le k,j \le N}\\ \mathbf{F_l} &= (f_0^l, f_1^l, \ldots, f_N^l,),\end{aligned} \tag{3.1.15}$$

where $a^l_{kj} = (D^{\nu}_{t_{l-1}} P^{(\alpha,\beta)}_{\Omega_l,j}, P^{(\alpha,\beta)}_{\Omega_l,k})_{W^{(\alpha,\beta)}_{\Omega_l}}$ $(0 \le k \le N-m,\ 0 \le j \le N)$, $a_{kj} = D^{k-N+m-1} P^{(\alpha,\beta)}_{\Omega_l,j}(t_{l-1})$ $(N-m+1 \le k \le N,\ 0 \le j \le N)$, $f_k^l = \Big(g(x), P^{(\alpha,\beta)}_{\Omega_l,k}(x) \Big)_{W^{(\alpha,\beta)}_{\Omega_l}}$ $(k = 0, 1, \ldots, N-m)$, $f_k^l = D^{(k-N+m-1)} u_{l-1,N}(t_{l-1})$ $(k = N-m+1, \ldots, N)$. We investigate that (3.1.13) (3.1.14) are equivalent to the matrix system

$$\mathbf{A_l K_l} = \mathbf{F_l}.$$

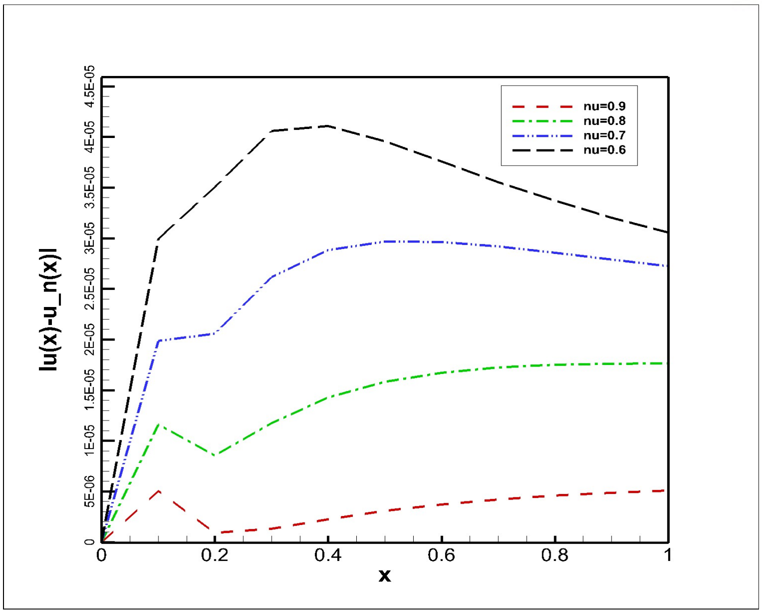

Figure 3-1: Numerical errors by using M-SJT(N of degree is 4 and n of step is 10) for **Example 1**($\nu = 0.6, 0.7, 0.8, 0.9$)

3.1.3 Numerical results of the linear FDE

In this section, we demonstrate numerical results of initial value problem by using M-SJT method. Our numerical results by M-SJT are compared with exact solutions.
Example 1. Consider the following Linear initial value problem.

$$D^{\nu}u(x) = \frac{\Gamma(5)}{\Gamma(5-\nu)}x^{4-\nu} - \frac{\Gamma(2.5)}{\Gamma(2.5-\nu)}x^{1.5-\nu}, \qquad u(0) = 0, \qquad x \in [0, 1) \tag{3.1.16}$$

having an exact solution given by $u(x) = x^4 - x^{1.5}$.

Figure 3-1 is Numerical results by using M-SJT(N of degree is 4 and n of step is 10) of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 1**($\nu = 0.6, 0.7, 0.8, 0.9$). If ν is close to 1, we can obtain a more accurate results because of smaller the proportion of the memory term. **Table 3-1** and **Table 3-2** are maximum errors by M-SJT for **Example 1**($\nu = 0.6, 0.7, 0.8$). **Table 3-3** and **Table 3-4** are Rates of convergence of n(step size) by M-SJT for **Example 1**($\nu = 0.6, 0.7, 0.8, 0, 9$). **Table 3-5** and **Table 3-6** are Rates of convergence of N(degree) by M-SJT for **Example 1**($\nu = 0.6, 0.7, 0.8$). We can know the maximum errors and rate of convergence of N(degree) and rate of convergence of n(step size) for the changes in each of the n and N. M-SJT has a better accuracy than SJT. Also, rate of convergence of n(step size) is approximately 1.5 and rate of convergence of N(degree) is approximately 3.

Table 3-1: Maximum errors by using M-SJT of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	0.101925	0.00162805	0.000213041
	n=2	0.0421765	0.000575604	7.53212E-05
	n=4	0.00718316	0.000203507	2.66301E-05
	n=8	0.00172708	7.19505E-05	9.42E-06
	n=16	0.000784032	2.54383E-05	3.33E-06
$\nu = 0.7$	n=1	0.105423	0.00167809	0.000216648
	n=2	0.043976	0.000593295	7.65966E-05
	n=4	0.0100489	0.000209762	2.7081E-05
	n=8	0.00158909	7.41619E-05	9.57E-06
	n=16	0.00016913	2.62202E-05	3.39E-06
$\nu = 0.8$	n=1	0.108688	0.00173075	0.000220358
	n=2	0.0425629	0.000611911	7.79084E-05
	n=4	0.0115164	0.000216343	2.75448E-05
	n=8	0.00278116	7.64889E-05	9.74E-06
	n=16	0.000593474	2.70429E-05	3.44E-06

Table 3-2: Maximum errors by using M-SJT of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	0.19178	0.00317445	0.000376435
	n=2	0.0534441	0.00112234	0.00013309
	n=4	0.00854599	0.000396806	4.70544E-05
	n=8	0.00145001	0.000140292	1.66E-05
	n=16	0.000714023	4.96008E-05	5.88E-06
$\nu = 0.7$	n=1	0.217548	0.00359122	0.000407103
	n=2	0.0580026	0.00126969	0.000143933
	n=4	0.0120463	0.000448903	5.08878E-05
	n=8	0.00205231	0.000158711	1.80E-05
	n=16	0.000482736	5.61128E-05	6.36E-06
$\nu = 0.8$	n=1	0.250894	0.00413605	0.000461395
	n=2	0.0605137	0.00146231	0.000163128
	n=4	0.014462	0.000517006	5.76744E-05
	n=8	0.00353711	0.000182789	2.04E-05
	n=16	0.000816317	6.46258E-05	7.21E-06

Table 3-3: Rates of convergence of n(step size) by using M-SJT of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-	-
	n=2	1.272996673	1.499996485	1.500003216
	n=4	2.553748823	1.499998198	1.499998139
	n=8	2.056283739	1.500001802	1.500001861
	n=16	1.139350468	1.500002452	1.499999222
$\nu = 0.7$	n=1	-	-	-
	n=2	1.261401363	1.499998563	1.500000659
	n=4	2.129678799	1.499996278	1.499999341
	n=8	2.660764845	1.500003235	1.499999905
	n=16	3.231996345	1.499999516	1.499997964
$\nu = 0.8$	n=1	-	-	-
	n=2	1.352524308	1.500003609	1.499998482
	n=4	1.885906647	1.500001392	1.499998899
	n=8	2.049931154	1.499998136	1.500001101
	n=16	2.228430013	1.50000053	1.499998899

Table 3-4: Rates of convergence of n(step size) by using M-SJT of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-	-
	n=2	1.843349681	1.4999968675	1.499998597
	n=4	2.644711155	1.500004044	1.500000636
	n=8	2.559184779	1.500001097	1.5000037
	n=16	1.022020398	1.499997448	1.4999963
$\nu = 0.7$	n=1	-	-	-
	n=2	1.907144277	1.499997736	1.499996451
	n=4	2.267527478	1.500000657	1.500005675
	n=8	2.553269523	1.500001615	1.49999633
	n=16	2.087942341	1.500000313	1.500002536
$\nu = 0.8$	n=1	-	-	-
	n=2	2.051744266	1.500004441	1.499997936
	n=4	2.064994717	1.499996257	1.500001438
	n=8	2.03162409	1.50000177	1.499998562
	n=16	2.115369679	1.499997114	1.500001438

Table 3-5: Rates of convergence of N(degree) by using M-SJT of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	5.968219139	2.933941997
	n=2	-	6.195218951	2.933948728
	n=4	-	5.141468326	2.933948668
	n=8	-	4.585186389	2.933948728
	n=16	-	4.945838373	2.933945498
$\nu = 0.7$	n=1	-	5.973225749	2.95339527
	n=2	-	6.21182295	2.953397365
	n=4	-	5.582140429	2.953400428
	n=8	-	4.42137882	2.953397098
	n=16	-	2.68938199	2.953395546
$\nu = 0.8$	n=1	-	5.972651506	2.973476169
	n=2	-	6.120130807	2.973471042
	n=4	-	5.734225552	2.973468549
	n=8	-	5.184292535	2.973471514
	n=16	-	4.455863052	2.973469883

Table 3-6: Rates of convergence of N(degree) by using M-SJT of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 1**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	5.91680181	3.076033977
	n=2	-	5.573448994	3.07603571
	n=4	-	4.428741883	3.076032301
	n=8	-	3.369558202	3.076034904
	n=16	-	3.847535252	3.076033756
$\nu = 0.7$	n=1	-	5.920715908	3.141008278
	n=2	-	5.513569366	3.141006993
	n=4	-	4.746042545	3.141012011
	n=8	-	3.692774638	3.141006726
	n=16	-	3.10483261	3.141008949
$\nu = 0.8$	n=1	-	5.922680533	3.164179351
	n=2	-	5.370940708	3.164172847
	n=4	-	4.805942248	3.164178028
	n=8	-	4.274319927	3.16417482
	n=16	-	3.658947362	3.164179144

Example 2. Consider the following Linear initial value problem with Mittag-Leffler type .

$$D^2u(x) + D^{\nu}u(x) + u(x) = f(x), \tag{3.1.17}$$

$$u(0) = 0 \text{ and } u'(0) = 1, \qquad x \in [0, 1) \tag{3.1.18}$$

having an exact solution given by $u(x) = sin(x)$.

$$f(x) = -\frac{1}{2}i(i)^m x^{m-\nu}(E_{1,m-\nu+1}(ix) - (-1)^m E_{1,m-\nu+1}(-ix)).$$

where $E_{\alpha,\beta}(z)$ is the two-parameter function of Mittag-Leffler type.
Table 3-7 and **Table 3-8** are maximum errors by M-SJT for **Example 2**($\nu = 1.3, 1.5, 1.7$). **Table 3-9** and **Table 3-10** are Rates of convergence of n(step size) by M-SJT for **Example 2**($\nu = 1.3, 1.5, 1.7$). We can know the maximum errors and rate of convergence of n(step size) for the changes in each of the n and N. M-SJT has a better accuracy than SJT. Also, rate of convergence of n(step size) is approximately 4. In this case, we chose the N from 3 to 5 because maximum errors are sufficiently small.

Table 3-7: Maximum errors by using M-SJT of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 2**($\nu = 1.3, 1.5, 1.7$)

ν		N=3	N=4	N=5
$\nu = 1.3$	n=1	0.000909975	6.98617E-05	1.41E-06
	n=2	0.000401965	3.88788E-05	2.02E-06
	n=4	0.000127741	9.66E-06	1.80E-07
	n=8	3.61897E-05	1.62E-06	1.24E-08
	n=16	9.81E-06	2.37E-07	7.82E-10
$\nu = 1.5$	n=1	0.000935661	6.78393E-05	1.38E-06
	n=2	0.000372029	3.88788E-05	1.85E-06
	n=4	0.000123066	9.39E-06	1.66E-07
	n=8	3.61763E-05	1.61E-06	1.11E-08
	n=16	1.01256E-05	2.38E-07	6.41E-10
$\nu = 1.7$	n=1	0.000997834	6.67346E-05	1.39E-06
	n=2	0.000299937	3.31541E-05	1.64E-06
	n=4	0.00010463	8.54E-06	1.62E-07
	n=8	3.23564E-05	1.55E-06	1.17E-08
	n=16	9.46E-06	2.39E-07	7.03E-10

Table 3-8: Maximum errors by using M-SJT of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 2**($\nu = 1.3, 1.5, 1.7$)

ν		N=3	N=4	N=5
$\nu = 1.3$	n=1	0.00333941	0.000320178	4.78E-06
	n=2	0.00071055	4.19461E-05	2.12E-06
	n=4	0.00014964	9.35E-06	1.82E-07
	n=8	3.75889E-05	1.60E-06	1.25E-08
	n=16	9.90E-06	2.36E-07	7.85E-10
$\nu = 1.5$	n=1	0.0033487	0.000324282	4.76E-06
	n=2	0.000687266	4.03369E-05	1.94E-06
	n=4	0.000145457	9.09E-06	1.68E-07
	n=8	3.76058E-05	1.58E-06	1.11E-08
	n=16	1.02139E-05	2.36E-07	6.44E-10
$\nu = 1.7$	n=1	0.0034996	0.000345504	4.98E-06
	n=2	0.000638091	3.68422E-05	1.74E-06
	n=4	0.000128837	8.33E-06	1.64E-07
	n=8	3.39092E-05	1.53E-06	1.17E-08
	n=16	9.56E-06	2.38E-07	7.06E-10

Table 3-9: Rates of convergence of n(step size) by using M-SJT of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 2**($\nu = 1.3, 1.5, 1.7$)

ν		N=3	N=4	N=5
$\nu = 1.3$	n=1	-	-	-
	n=2	1.178757022	0.845518059	-0.526763828
	n=4	1.653849908	2.00908906	3.494889841
	n=8	1.819568926	2.574711052	3.854260435
	n=16	1.883327921	2.772616792	3.990048234
$\nu = 1.5$	n=1	-	-	-
	n=2	1.330570836	0.803137593	-0.417760363
	n=4	1.595982848	2.050107774	3.474956441
	n=8	1.766315471	2.544290038	3.901598397
	n=16	1.837037463	2.759488329	4.113998355
$\nu = 1.7$	n=1	-	-	-
	n=2	1.734140325	1.00924766	-0.241337367
	n=4	1.519362936	1.956156972	3.345112225
	n=8	1.693173128	2.459534972	3.785905229
	n=16	1.773698628	2.697365138	4.058882864

Table 3-10: c by using M-SJT of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$)) for **Example 2**($\nu = 1.3, 1.5, 1.7$)

ν		N=3	N=4	N=5
$\nu = 1.3$	n=1	-	-	-
	n=2	2.232585155	2.932265596	1.174546841
	n=4	2.247440302	2.165495324	3.545041062
	n=8	1.993117269	2.548828442	3.866253577
	n=16	1.925368882	2.759539018	3.98778636
$\nu = 1.5$	n=1	-	-	-
	n=2	2.284660641	3.007076828	1.293567627
	n=4	2.240275862	2.149827372	3.531573388
	n=8	1.951565633	2.520464667	3.914596881
	n=16	1.880421349	2.746125787	4.113701382
$\nu = 1.7$	n=1	-	-	-
	n=2	2.455355943	3.229271295	1.515953894
	n=4	2.308215213	2.144214159	3.412121894
	n=8	1.92579832	2.445541496	3.799792048
	n=16	1.826776837	2.685122669	4.055631576

3.2 Model Problem

Let us consider the nonlinear initial value problem of fractional differential equation as follows

$$D^{\nu}u(x) = F\Big(x, u(x)\Big), \qquad \text{in } \Omega = (0,1),$$
$$u^{(i)}(0) = d_i, \qquad i = 0, ..., m-1, \tag{3.2.1}$$

where $m-1 < \nu \leqslant m$ and $m \in \mathbb{N}$. Here, the fractional derivative $D^{\nu}u(x)$ denotes the Caputo fractional derivative of order ν for $u(x)$, and the values of $d_i(i = 0, \cdots, m-1)$ describe the initial data of $u(x)$ and $F(x)$ is a given source function. We apply the shifted jacobi polynomials to the model (3.2.1) in obtaining an accurate approximation.

3.2.1 Shifted Jacobi collocation (SJC) method for initial value problem

In this section, we show the spectral shifted jacobi collocation method for the numerical solution of the nonlinear initial value problems(3.2.1). Suppose that $u_N(x)$ can be approximated by N-partial sum

$$u_N(x) = \sum_{i=0}^{N} \xi_i^l P_{\Omega,i}^{(\alpha,\beta)}(x).$$

The implementation of spectral shifted Jacobi collocation approximation for solving numerically (3.2.1) is to find$u_N(x) \in S_N(0,1)$ such that

$$D^{\nu}u_N(x) = F\Big(x, u_N(x)\Big), \qquad \text{in } \Omega = (0,1) \tag{3.2.2}$$

is exactly satisfied at $x_{\Omega),N,k}^{(\alpha,\beta)}, k = 0, 1, \dots, N-m$.Thus, we have to collocate (3.2.2) at the $(N-m+1)$ shifted Jacobi roots $x_{\Omega,N,k}^{(\alpha,\beta)}$,which gives

$$\sum_{j=0}^{N} \xi_j D^{\nu} P_{\Omega,i}^{(\alpha,\beta)}(x_{\Omega,N,k}^{(\alpha,\beta)}) = F\Big(x_{\Omega,N,k}^{(\alpha,\beta)}, \sum_{i=0}^{N} \xi_i^l P_{\Omega,i}^{(\alpha,\beta)}(x_{\Omega,N,k}^{(\alpha,\beta)})\Big) \tag{3.2.3}$$

with initial conditions of (3.2.1) written in the form

$$\sum_{j=0}^{N} \xi_j^l D^i P_{\Omega,j}^{(\alpha,\beta)}(0) = d_i, \qquad i = 0, ..., m-1. \tag{3.2.4}$$

The results constitute a system of$(N+1)$ nonlinear algebraic equations in $\xi_j(j = 0, 1, \dots, N)$, which can be evaluated immediately by implementing Newton's iteration method.

3.2.2 Mutistage shifted Jacobi collocation (M-SJC) method for initial value problem

In the previous section we described the basic idea of the SJC method. In the SJC method, it is the key to solve the equations (3.2.3),(3.2.4) by implementing Newton's iteration method. Also it is cleat that the equations become larger if the solution is approximated with many shifted jacobi polynomial. That is, N is large in (3.2.3),(3.2.4). In this section, we propose an efficient computational method, namely Multistage Shifted Jacobi Collocation(M-SJC) method. The basic idea of the M-SJC method is to apply the standard SJC method to the problem in each sub-domain. To describe the method, we consider the equally spaced partition $P : 0 = x_0 < x_1 < \cdots < x_n = 1$, where the nodes $x_l = l * h, h = 1/n, l = 0, \cdots, n$. Let us define the lth sub-domain $\Omega_l \equiv (x_{l-1}, x_l)$ and $u(x) \mid_{(\Omega_l)} \equiv u_l(x)$. We recall that $\{P^{(\alpha,\beta)}_{\Omega,n}(x) : n \geq 0\}$ forms a complete orthogonal system in $L^2_{W^{(\alpha,\beta)}_{\Omega}}$,where

$$P^{(\alpha,\beta)}_{(\Omega_l,n}(x) = \begin{cases} \sum_{k=0}^{i} \frac{\Gamma(i+\alpha+1)\Gamma(i+k+\alpha+\beta+1)}{\Gamma(k+\alpha+1)\Gamma(i+\alpha+\beta+1)(i-k)!k!(b-a)^k}(x-b)^k, & \text{if } x \in \Omega_l, \\ 0, & \text{if } x \notin \Omega_l. \end{cases}$$

Suppose that all solutions $u_i(x), i = 1, \cdots, l-1$, are approximated by $u_{i,N}(x)$, where $u_{i,N}(x) = \sum_{j=0}^{N} \xi^i_j P^{(\alpha,\beta)}_{\Omega_i,j}(x)$. Now, let us define

$$S_N(\Omega_l) = \text{span}\{1, P^{(\alpha,\beta)}_{\Omega_l,1}(x), P^{(\alpha,\beta)}_{\Omega_l,2}(x), \ldots, P^{(\alpha,\beta)}_{\Omega_l,N}(x)\}.$$

and suppose that an approximation $u_{l,N}(x) \in S_N(\Omega_l)$ of $u_l(x)$ is defined by

$$u_{l,N}(x) = \begin{cases} \sum_{i=0}^{N} \xi^l_i P^{(\alpha,\beta)}_{\Omega_l,i}(x), & \text{if } x \in \Omega_l, \\ 0, & \text{if } x \notin \Omega_l \end{cases}.$$

For $x \in \Omega_l$, we have

$$\begin{aligned} D^{\nu}_0 u(x) &= \int_{x_0}^{x} (x-s)^{m-\nu-1} D^m u(s) ds \\ &= \sum_{k=0}^{l-1} \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_k(s) ds + \int_{x_{l-1}}^{x} (x-s)^{m-\nu-1} D^m u_l(s) ds \\ &\approx \sum_{k=0}^{l-1} \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_{k,N}(s) ds + \int_{x_{l-1}}^{x} (x-s)^{m-\nu-1} D^m u_{l,N}(s) ds. \end{aligned} \tag{3.2.5}$$

Let us define

$$D^{\nu}_{\Omega_k} u_{k,N}(x) = \int_{x_{k-1}}^{x_k} (x-s)^{m-\nu-1} D^m u_{k,N}(s) ds. \tag{3.2.6}$$

Since

$$D_0^{\nu} u(x) = F(x, u(x)), \quad \text{in } (x_0, x_l), \tag{3.2.7}$$

we have for $x \in (x_{l-1}, x_l)$,

$$D_{\Omega_l}^{\nu} u_l(x) = F(x, u(x)) - \sum_{k=0}^{l-1} D_{\Omega_k}^{\nu} u_k(x). \tag{3.2.8}$$

The standard SJC method to (3.2.8) is to obtain $u_{l,N}(x)$ which satisfies the following

$$D_{\Omega_l}^{\nu} u_{l,N}(x) = F(x, u_{l,N}(x)) - \sum_{k=0}^{l-1} D_{\Omega_k}^{\nu} u_{k,N}(x). \tag{3.2.9}$$

is satisfied exactly at $x_{\Omega_l,N,h}^{(\alpha,\beta)}(h = 0, 1, \cdots, N - m)$. Thus, we have to collocate (3.2.9) at the $(N - m + 1)$ shifted Jacobi roots $x_{\Omega_l,N,h}^{(\alpha,\beta)}$,which gives

$$\sum_{j=0}^{N} \xi_j D^{\nu} P_{\Omega_l,j}^{(\alpha,\beta)}(x_{(\Omega_l),N,h}^{(\alpha,\beta)}) = F\Big(x_{\Omega_l,N,h}^{(\alpha,\beta)}, \sum_{i=0}^{N} \xi_i^l P_{\Omega_l,i}^{(\alpha,\beta)}(x_{\Omega_l,N,h}^{(\alpha,\beta)})\Big) - \sum_{k=0}^{l-1} D_{\Omega_k}^{\nu} P_{\Omega_k,i}^{(\alpha,\beta)}(x_{\Omega_k,N,h}^{(\alpha,\beta)})$$

with initial conditions of (3.2.1) written in the form

$$\sum_{j=0}^{N} \xi_i^l D^i P_{\Omega_l,j}^{(\alpha,\beta)}(t_{l-1}) = D^i u_{l-1,N}(t_{l-1}), \quad i = 0, ..., m - 1.$$

The results constitute a system of$(N + 1)$ nonlinear algebraic equations in $\xi_j^l (j = 0, 1, \ldots, N)$, which can be evaluated immediately by implementing Newton's iteration method.

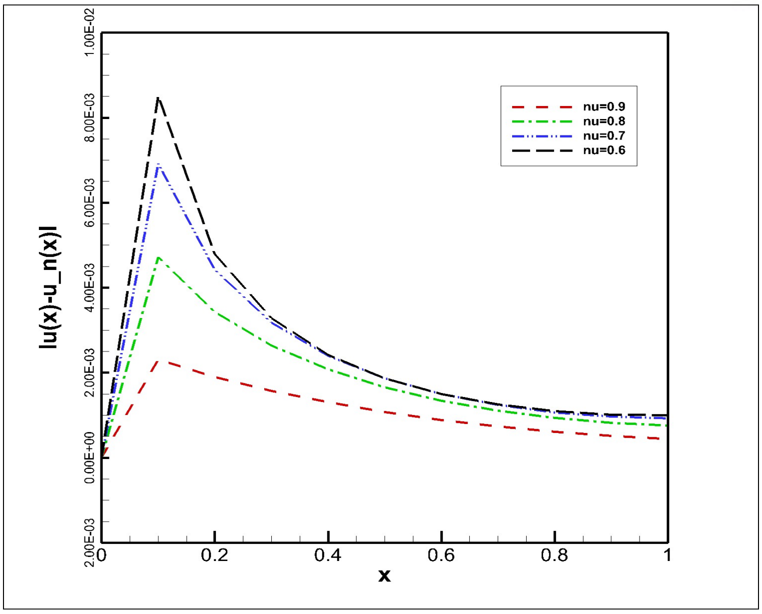

Figure 3-2: Numerical errors by using M-SJC(N of degree is 4 and n of step is 10) for **Example 3**($\nu = 0.6, 0.7, 0.8, 0.9$)

3.2.3 Numerical results of the nonlinear FDE

In this section, we demonstrate numerical results of initial value problem by using M-SJC method. Our numerical results by M-SJC are compared with exact solutions.

Example 3. Consider the following nonlinear initial value problem.

$$D^{\nu}u(x) = \frac{\Gamma(9)}{\Gamma(9-\nu)}x^{8-\nu} - 3\frac{\Gamma(5+\frac{\nu}{2})}{\Gamma(5-\frac{\nu}{2})}x^{4-\frac{\nu}{2}} + \frac{9}{4}\Gamma(1+\nu) + (\frac{3}{2}x^{\frac{\nu}{2}} - x^4)^3 - (u(x))^{\frac{3}{2}},$$
$$u(0) = 0, \qquad x \in [0, 1) \tag{3.2.10}$$

having an exact solution given by $u(x) = x^8 - 3x^{4+\frac{\nu}{2}} + \frac{9}{4}x^{\nu}$.

Figure 3-2 is Numerical results by using M-SJC(N of degree is 4 and n of step is 10) of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 3**($\nu = 0.6, 0.7, 0.8, 0.9$). If ν is close to 1, we can obtain a more accurate results because of smaller the proportion of the memory term. **Table 3-11** and **Table 3-12** are maximum errors by M-SJC for **Example 3**($\nu = 0.6, 0.7, 0.8$). **Table 3-13** and **Table 3-14** are Rates of convergence of n(step size) by M-SJC for **Example 3**($\nu = 0.6, 0.7, 0.8$). **Table 3-15** and **Table 3-16** are Rates of convergence of N(degree) by M-SJC for **Example 3**($\nu = 0.6, 0.7, 0.8$). We can know the maximum errors and rate of convergence of N(degree) and rate of convergence of n(step size) for the changes in each of the n and N. M-SJC has a better accuracy than SJC. Also, rate of convergence of n(step size) and rate of convergence of N(degree) depend on ν.

Table 3-11: Maximum errors by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	0.122551	0.177986	0.0752457
	n=2	0.153747	0.0969701	0.0510395
	n=4	0.117272	0.0662499	0.0320982
	n=8	0.080874	0.0446439	2.28E-02
	n=16	0.0546024	0.0298094	1.51E-02
$\nu = 0.7$	n=1	0.149412	0.145856	0.0430404
	n=2	0.0881948	0.0577757	0.0272856
	n=4	0.071374	0.0370495	0.0170503
	n=8	0.0467496	0.023298	1.06E-02
	n=16	0.0295066	0.0144927	6.53E-03
$\nu = 0.8$	n=1	0.251052	0.139622	0.0225811
	n=2	0.0380653	0.0315885	0.0133401
	n=4	0.0392493	0.018958	0.00775954
	n=8	0.0247159	0.0111097	4.48E-03
	n=16	0.0145713	0.00643528	2.58E-03

Table 3-12: Maximum errors by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	0.124199583	1.72E-01	7.09E-02
	n=2	0.153745374	9.44E-02	4.79E-02
	n=4	1.18E-01	6.44E-02	3.21E-02
	n=8	8.17E-02	4.33E-02	2.14E-02
	n=16	5.52E-02	2.89E-02	1.42E-02
$\nu = 0.7$	n=1	0.141086767	1.39E-01	4.01E-02
	n=2	0.087346879	5.60E-02	2.53E-02
	n=4	7.20E-02	3.58E-02	1.58E-02
	n=8	4.73E-02	2.25E-02	9.79E-03
	n=16	2.99E-02	1.40E-02	6.05E-03
$\nu = 0.8$	n=1	0.239122516	1.31E-01	2.08E-02
	n=2	0.036663665	3.05E-02	1.22E-02
	n=4	3.95E-02	1.82E-02	7.11E-03
	n=8	2.50E-02	1.07E-02	4.10E-03
	n=16	1.48E-02	6.17E-03	2.36E-03

Table 3-13: Rates of convergence of n(step size) by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-	-
	n=2	-0.327176004	0.87615189	0.559994944
	n=4	0.390699665	0.549621695	0.6691218
	n=8	0.536110721	0.569455213	0.49424974
	n=16	0.566711603	0.582695726	0.592777188
$\nu = 0.7$	n=1	-	-	-
	n=2	0.760530521	1.336009995	0.657551717
	n=4	0.305294969	0.641008761	0.678342645
	n=8	0.610444606	0.66924796	0.689275892
	n=16	0.66391633	0.684879718	0.694756688
$\nu = 0.8$	n=1	-	-	-
	n=2	2.721437865	2.144054946	0.759346285
	n=4	-0.044190476	0.736592658	0.781726447
	n=8	0.667227483	0.770986914	0.791915392
	n=16	0.762309846	0.787745033	0.796455424

Table 3-14: Rates of convergence of n(step size) by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-	-
	n=2	-0.307882666	0.865864545	0.563659008
	n=4	0.37814606	0.551878124	0.57876807
	n=8	0.534518297	0.570866575	0.588127263
	n=16	0.566260639	0.583514278	0.593492336
$\nu = 0.7$	n=1	-	-	-
	n=2	0.691754626	1.316211386	0.66212179
	n=4	0.27844571	0.644307854	0.680721096
	n=8	0.607126417	0.670906512	0.690467838
	n=16	0.66318789	0.685704415	0.695343606
$\nu = 0.8$	n=1	-	-	-
	n=2	2.705327067	2.109829733	0.764308116
	n=4	-0.109139537	0.741043101	0.783989644
	n=8	0.659937568	0.772790122	0.792935259
	n=16	0.76118452	0.788523565	0.796905149

Table 3-15: Rates of convergence of N(degree) by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-0.53838151	1.242082721
	n=2	-	0.664946383	0.925925776
	n=4	-	0.823868413	1.045425881
	n=8	-	0.857212905	0.970220407
	n=16	-	0.873197028	0.980301869
$\nu = 0.7$	n=1	-	0.034751288	1.760781344
	n=2	-	0.610230762	1.082323066
	n=4	-	0.945944555	1.119656949
	n=8	-	1.004747908	1.139684881
	n=16	-	1.025711296	1.149561851
$\nu = 0.8$	n=1	-	0.846459936	2.628338611
	n=2	-	0.269077017	1.24362995
	n=4	-	1.049860151	1.288763739
	n=8	-	1.153619582	1.309692218
	n=16	-	1.179054769	1.318402609

Table 3-16: Rates of convergence of N(degree) by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 3**($\nu = 0.6, 0.7, 0.8$)

ν		N=2	N=4	N=8
$\nu = 0.6$	n=1	-	-0.4698905	1.279558406
	n=2	-	0.703856712	0.977352868
	n=4	-	0.877588775	1.004242815
	n=8	-	0.913937053	1.021503503
	n=16	-	0.931190693	1.03148156
$\nu = 0.7$	n=1	-	0.017399075	1.797821819
	n=2	-	0.641855835	1.143732222
	n=4	-	1.007717979	1.180145465
	n=8	-	1.071498074	1.19970679
	n=16	-	1.094014598	1.209345982
$\nu = 0.8$	n=1	-	0.863066266	2.660911973
	n=2	-	0.267568931	1.315390357
	n=4	-	1.11775157	1.3583369
	n=8	-	1.230604124	1.378482037
	n=16	-	1.25794317	1.38686362

Table 3-17: Maximum errors by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 4**

Maximum errors		N=2	N=4	N=8
Legendre	n=1	0.319709	0.15983	0.141623
	n=2	5.99E-03	2.99E-04	1.89E-05
	n=4	1.31E-03	6.40E-05	3.96E-06
	n=8	2.80E-04	1.35E-05	8.18E-07
	n=16	5.88E-05	2.79E-06	1.68E-07

Table 3-18: Maximum errors by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 4**

Maximum errors		N=2	N=4	N=8
Chebyshev	n=1	0.443608	0.171602	0.073748
	n=2	6.26E-03	2.73E-04	1.53E-05
	n=4	1.37E-03	5.83E-05	3.19E-06
	n=8	2.93E-04	1.22E-05	6.59E-07
	n=16	6.15E-05	2.54E-06	1.35E-07

Example 4. Consider the following nonlinear initial value problem.

$$D^{0.7}u(x) = \begin{cases} \frac{\Gamma(2.3+1)}{\Gamma(2.3+1-0.7)}x^{1.6} + \mid x^{3-0.7} - \frac{1}{2}^{3-0.7} \mid -u(x), & \text{if } x \le \frac{1}{2}, \\ f(x) + \mid x^{3-0.7} - \frac{1}{2}^{3-0.7} \mid -u(x), & \text{if } x > \frac{1}{2} \end{cases} \quad x \in [0,1),$$
$$u(0) = \frac{1}{2}^{3-0.7} \tag{3.2.11}$$

having an exact solution given by $u(x) = \mid x^{3-0.7} - \frac{1}{2}^{3-0.7} \mid$

$$f(x) = \frac{2.3}{\Gamma(0.3)} {}_2F_1(0.7, 2.3; 3.3; 0.5t)\Big(2.44141t^{1.6} - \frac{0.0882883}{t^{0.7}} - \frac{0.0882883(1-0.5t^{-1})^{0.7}}{(-0.5+t)^{0.7}}\Big).$$

${}_2F_1(q, b; c; z)$ is Hypergeometric function. **Table 3-17** and **Table 3-18** are maximum errors by M-SJC for **Example 4**. **Table 3-19** and **Table 3-20** are rates of convergence of n(step size) by M-SJC for **Example 4**. **Table 3-21** and **Table 3-22** arerates of convergence of N(degree) by M-SJC for **Example 4**. We can know the maximum errors and rate of convergence of N(degree) and rate of convergence of n(step size) for the changes in each of the n and N. M-SJC has a much better accuracy than SJC. This example is a special case Because it is not smooth at 0.5. In this case, we should take advantage of the M-SJC. Also, rate of convergence of n(step size) is approximately 2.3 and rate of convergence of N(degree) is approximately 4.

Table 3-19: Rates of convergence of n(step size) by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 4**

Rates of n		N=2	N=4	N=8
Legendre	n=1	-	-	-
	n=2	5.73769109	9.060007379	12.86899448
	n=4	2.194404254	2.225148273	2.258659152
	n=8	2.226181245	2.250641648	2.273843223
	n=16	2.250739711	2.268292088	2.283605954

Table 3-20: Rates of convergence of n(step size) by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 4**

Rates of n		N=2	N=4	N=8
Chebyshev	n=1	-	-	-
	n=2	6.147048056	9.294681955	12.2344323
	n=4	2.193740572	2.227732584	2.261762375
	n=8	2.22559289	2.252442444	2.27586978
	n=16	2.250297285	2.269492871	2.284828718

Table 3-21: Rates of convergence of N(degree) by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 4**

Rates of N		N=2	N=4	N=8
Legendre	n=1	-	1.000221131	0.174482644
	n=2	-	4.32253742	3.983469744
	n=4	-	4.353281439	4.016980623
	n=8	-	4.377741841	4.040182198
	n=16	-	4.395294217	4.055496064

Table 3-22: Rates of convergence of N(degree) by using M-SJC of Chebyshev polynomials($\alpha = -1/2$ and $\beta = -1/2$) for **Example 4**

Rates of N		N=2	N=4	N=8
Chebyshev	n=1	-	1.370219016	1.218390537
	n=2	-	4.517852914	4.15814088
	n=4	-	4.551844926	4.192170671
	n=8	-	4.57869448	4.215598007
	n=16	-	4.597890066	4.230933854

IV

Boundary Value Problem of Fractional Differential Equation

4.1 Model Problem

Let us consider the nonlinear boundary value problem of fractional differential equation as follows

$$\begin{aligned} & D^{2\nu}u(x) = F\Big(x, u(x), D^{\nu}u(x)\Big), \qquad \text{in } \Omega = (0,1), \\ & u(0) = d_0,\ u(1) = d_1, \end{aligned} \tag{4.1.1}$$

where $0.5 < \nu < 1$ and $m-1 < \nu \leqslant m$ and $m \in \mathbb{N}$. Here, the fractional derivative $D^{2\nu}u(x)$ denotes the Caputo fractional derivative of order ν for $u(x)$, and the values of $d_i (i = 0, 1)$ describe the initial and boundary data of $u(x)$ and $F(x)$ is a given source function. We apply the shifted jacobi polynomials to the model (4.1.1) in obtaining an accurate approximation.

4.2 Shifted Jacobi collocation (SJC) method for boundary value problem

In this section, we show the spectral shifted jacobi collocation method for the numerical solution of the nonlinear boundary value problems(4.1.1). Suppose that $u_N(x)$ can be approximated by N-partial sum

$$u_N(x) = \sum_{i=0}^{N} \xi_i^l P_{\Omega,i}^{(\alpha,\beta)}(x).$$

The implementation of spectral shifted Jacobi collocation approximation for solving numerically (4.1.1) is to find $u_N(x) \in S_N(\Omega)$ such that

$$D^{2\nu}u_N(x) = F\Big(x, u_N(x), D^{\nu}u_N(x)\Big), \quad \text{in } \Omega = (0,1) \tag{4.2.1}$$

is exactly satisfied at $x^{(\alpha,\beta)}_{\Omega,N,k}, k = 0, 1, \ldots, N - m$. We have to collocate (4.2.1) at the $(N - m + 1)$ shifted Jacobi roots $x^{(\alpha,\beta)}_{\Omega,N,k}$,which gives

$$\sum_{j=0}^{N} \xi_j D^{2\nu} P^{(\alpha,\beta)}_{\Omega,i}(x^{(\alpha,\beta)}_{\Omega,N,k}) = F\Big(x^{(\alpha,\beta)}_{\Omega,N,k}, \sum_{i=0}^{N} \xi_i^l P^{(\alpha,\beta)}_{\Omega,i}(x^{(\alpha,\beta)}_{\Omega,N,k}), \sum_{i=0}^{N} \xi_i D^{\nu} P^{(\alpha,\beta)}_{\Omega,i}(x^{(\alpha,\beta)}_{\Omega,N,k})\Big)$$

with (4.1.1) written in the form

$$\sum_{j=0}^{N} \xi_j^l P^{(\alpha,\beta)}_{\Omega,j}(0) = d_0 \quad \text{and} \quad \sum_{j=0}^{N} \xi_j^l P^{(\alpha,\beta)}_{\Omega,j}(1) = d_1.$$

The results constitute a system of$(N + 1)$ nonlinear algebraic equations in $\xi_j (j = 0, 1, \ldots, N)$, which can be evaluated immediately by implementing Newton's iteration method.

4.3 Multistage Shifted Jacobi collocation (M-SJC) method for boundary value problem

Mutistage shifted Jacobi collocation (M-SJC) method can't calculate boundary value problem. However, we use Shooting method in the section 4.3.1 so that we chage the boundary value problem to initial value problem. Then, we use Mutistage shifted Jacobi collocation (M-SJC) method in the section 3.2.2.

4.3.1 Shooting method for the boundry value problem

In this section, We discuss a new numerical scheme for solving the equation (4.1.1). We have not got any information about $D^{\nu}u(0)$, we apply shooting method to this problem.

$$\begin{aligned} &D^{2\nu}u(x) = f\Big(x, u(x), D^{\nu}u(x)\Big), \quad \text{in } \Omega = (0,1) \\ &u(0) = d_0, \ D^{\nu}u(x) = s. \end{aligned} \tag{4.3.1}$$

View s as a variable, we can get $u(1, s) =: u(s)$ when s differs, But since Neton's Method is for the integer-order system problems, we have to adjust it for fractional-order cases. Define

$$F(s) = u(s) - d1.$$

Common Newton's Method is

$$s_{k+1} = s_k + \frac{F(s_k)}{F'(s_k)} \quad k = 0, 1, 2, \ldots.$$

We want to get

$$F'(s) = \frac{\partial u(1, s)}{\partial s}.$$

So we apply operator $\frac{\partial}{\partial s}$ on (4.3.1) that

$$\begin{aligned} &D^{2\nu} \frac{\partial u(x)}{\partial s} = \frac{\partial f\Big(x, u(x), D^{\nu} u(x)\Big)}{\partial s}, \qquad \text{in } \Omega = (0,1) \\ &\frac{\partial u(x)}{\partial s} \mid_{x=0} = 0, \ \frac{\partial D^{\nu} u(x)}{\partial s} \mid_{x=0} = 1. \end{aligned} \tag{4.3.2}$$

Notice x and s are unrelated, we can get

$$\frac{\partial f\Big(x, u(x), D^{\nu} u(x)\Big)}{\partial s} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial s} + \frac{\partial f}{\partial u}\frac{\partial u}{\partial s} + \frac{\partial f}{\partial D^{\nu} u}\frac{\partial D^{\nu} u}{\partial s}.$$

Define

$$\frac{\partial u}{\partial s} = \hat{u} \text{ and } \frac{\partial D^{\nu} u(x)}{\partial s} = D^{\nu} \hat{u}(x).$$

Then we can rewrite(4.3.2) as

$$\begin{aligned} &D^{2\nu} \hat{u}(x) = \frac{\partial f\Big(x, u(x), D^{\nu} u(x)\Big)}{\partial u} \cdot \hat{u}(x) + \frac{\partial f\Big(x, u(x), D^{\nu} u(x)\Big)}{\partial u'} \cdot D^{\nu} \hat{u}(x), \qquad \text{in } \Omega = (0,1) \\ &\hat{u}(0) = 0, \ D^{\nu} \hat{u}(0) = 1. \end{aligned} \tag{4.3.3}$$

We can get the clear from of $\frac{\partial f}{\partial u}$ and $\frac{\partial f}{\partial D^{\nu} u}$, Hence problem (4.3.3) can be solved by fractional-order methods as illustrated above. Once we get $\hat{u}(1)$, we can apply it to Neton's Method until $F(s)$ become small enough.

4.4 Numerical results of Fractional Boundary Value Differential Equation

In this section, we demonstrate numerical results of boundary value problem by using SJC and M-SJC methods. Our numerical results by SJC and M-SJC are compared with exact solutions.

Table 4-1: results by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 5**

ν	N=2	N=4	N=8
$\nu = 1.2$	0.030083	0.00217294	3.45463E-05
$\nu = 1.5$	0.0647222	0.00211785	5.37759E-05
$\nu = 1.8$	0.1244	0.00210102	5.11097E-05

Example 5. Consider the following nonlinear boundary value problem.

$$\begin{aligned} D^{\nu}u(x) &= \frac{\Gamma(5)}{\Gamma(5-\nu)}x^{4-\nu} - \frac{\Gamma(\frac{7}{2}+1)}{\Gamma(\frac{7}{2}+1-\nu)}x^{\frac{7}{2}-\nu} + (x^4 - x^{\frac{7}{2}})^2 - u^2(x) \\ &+ \frac{\Gamma(5)}{\Gamma(5-\frac{\nu}{2})}x^{4-\frac{\nu}{2}} - \frac{\Gamma(\frac{7}{2}+1)}{\Gamma(\frac{7}{2}+1-\frac{\nu}{2})}x^{\frac{7-\nu}{2}} - D^{\frac{\nu}{2}}u(x), \qquad (4.4.1) \\ & u(0) = 0 \text{ and } u(1) = 0 \end{aligned}$$

having an exact solution given by $u(x) = x^4 - x^{\frac{7}{2}}$.

Table 4-1 is maximum errors by SJC(N of degree) for **Example 5**($\nu = 1.2, 1.5, 1.8$). **Table 4-2, 3, 4**, are Sensitivity of initial s for **Example 5**($\nu = 1.2, 1.5, 1.8$) by M-SJC(N=4)with Shooting method. In the **Table 4-2, 3, 4**, reds are a Convergence value and blues are another Convergence value. We want to converge red value because exact initial value of s is zero in the **Example 5**. As n increases, the error is reduced and the sensitivity increases in the **Table 4-2, 3, 4**. If ν is 1.8 and s=15, we can see that convergence with a different initial condition(109.7398991). By considering the sensitivity in the **Table 4-2, 3, 4**, ex5 was solved by using M-SJC. **Table 4-5** is convergence s by using M-SJC with Shooting method of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 5**. **Table 4-6** is maximum errors by M-SJC(N of degree and n of step) with Shooting method for **Example 5**($\nu = 1.2, 1.5, 1.8$). In the results,we can know that M-SJC has a much better accuracy than SJC.

Table 4-2: Sensitivity of initial s for **Example 5**($\nu = 1.2, 1.5, 1.8$) by M-SJC(N=4 and n=1)with Shooting method.

ν	iteration	s=5	s=10	s=15	s=20
$\nu = 1.2$	m=1	-1.895921655	-6.124331069	-11.38045423	-17.09226091
	m=2	-0.320411327	-0.111954329	4.406490709	8.914886769
	m=3	0.005921082	0.015222899	-1.509844739	-5.090810098
	m=4	0.016793207	0.016804243	-0.203126682	-0.668057737
	m=5	0.016804481	0.016804481	0.012182889	-0.028195758
	m=6	0.016804481	0.016804481	0.016802448	0.016611612
	m=7	0.016804481	0.016804481	0.016804481	0.016804477
	m=8	0.016804481	0.016804481	0.016804481	0.016804481
$\nu = 1.5$	m=1	-2.179005329	-8.22529407	-17.41646537	-29.45686511
	m=2	-0.343770951	0.644316263	8.693777538	19.14378637
	m=3	0.013867068	-0.008742429	-6.323261645	-27.19914847
	m=4	0.025918199	0.025823826	-0.437151314	17.29427397
	m=5	0.025931208	0.025931207	0.007076668	-22.59701498
	m=6	0.025931208	0.025931208	0.025899441	13.35856487
	m=7	0.025931208	0.025931208	0.025931208	-14.07578305
	m=8	0.025931208	0.025931208	0.025931208	5.615632612
	m=9	0.025931208	0.025931208	0.025931208	-2.735329683
	m=10	0.025931208	0.025931208	0.025931208	-0.504173373
	m=11	0.025931208	0.025931208	0.025931208	0.00129747
	m=12	0.025931208	0.025931208	0.025931208	0.025876992
	m=13	0.025931208	0.025931208	0.025931208	0.025931208
$\nu = 1.8$	m=1	-2.39316892	-11.34761225	-31.58796542	-79.79574212
	m=2	-0.308558892	1.637909472	17.80377365	52.6165562
	m=3	0.02782717	-0.185770581	-52.8505112	134.7107735
	m=4	0.036913004	0.03310103	34.3694627	110.7683245
	m=5	0.036919483	0.03691834	314.4314945	109.7421045
	m=6	0.036919483	0.036919483	157.2438643	109.7398991
	m=7	0.036919483	0.036919483	112.9249099	109.7398991
	m=8	0.036919483	0.036919483	109.760539	109.7398991
	m=9	0.036919483	0.036919483	109.7398991	109.7398991

Table 4-3: Sensitivity of initial s for **Example 5**($\nu = 1.2, 1.5, 1.8$) by M-SJC(N=4 and n=2)with Shooting method.

ν	iteration	s=5	s=10	s=15	s=20
$\nu = 1.2$	m=1	-2.57238	-4.25063	-6.06821	-7.96137
	m=2	-0.386542	0.572081	8.46303	22.5841
	⋮	⋮	⋮	⋮	⋮
	m=8	0.00225383	0.00225464	0.000333445	-61.0522
	m=9	0.002254	0.002254	0.00243724	392.224
	m=10	0.002254	0.002254	0.00223646	-838.662
	m=11	0.002254	0.002254	0.00225566	-615.881
	m=12	0.002254	0.002254	0.002254	-362.679
$\nu = 1.5$	m=1	-2.65188	-4.78489	-7.28655	-10.0275
	m=2	-0.377868	0.00756519	6.39237	22.9743
	⋮	⋮	⋮	⋮	⋮
	m=6	0.00389743	0.00389794	0.00371198	634.812
	m=7	0.00389792	0.00389792	0.00390692	-166.744
	m=8	0.00389792	0.00389792	0.00389749	663.129
	m=9	0.00389792	0.00389792	0.00389795	-142.655
	m=10	0.00389792	0.00389792	0.00389792	643.19
$\nu = 1.8$	m=1	-2.75779	-5.69373	-9.77637	-15.0311
	m=2	-0.380164	-0.0230195	9.06858	41.7929
	m=3	-0.00216895	0.00740363	9.73467	-83.6456
	m=4	0.00734048	0.00728754	-11.5696	676.161
	m=5	0.00728827	0.00728828	17.8428	6159.67
	m=6	0.00728827	0.00728827	-41.6971	1048.72

Table 4-4: Sensitivity of initial s for **Example 5**($\nu = 1.2, 1.5, 1.8$) by M-SJC(N=4 and n=4)with Shooting method.

ν	iteration	s=5	s=7	s=9	s=11
$\nu = 1.2$	m=1	-2.86064	-4.64141	-6.55035	-8.52822
	m=2	-0.297197	4.36257	57.4065	212.768
	m=3	0.0306099	-2.33663	-56.8422	-205.188
	m=4	-0.00406562	-0.254542	-164326	-5634.72
	⋮	⋮	⋮	⋮	⋮
	m=9	0.000260423	0.000270571	-164326	-2799.13
	m=10	0.00026017	0.000258761	-164326	-2464.27
	m=11	0.00026017	0.0002604	-164326	-2200.83
	m=12	0.00026017	0.00026017	-164326	-1931.16
$\nu = 1.5$	m=1	-2.82732	-5.01967	-7.54775	-10.2785
	m=2	-0.364683	0.800001	17.6527	99.5935
	⋮	⋮	⋮	⋮	⋮
	m=7	0.000552505	0.000550071	3458.85	10661.9
	m=8	0.000552236	0.000552402	-9646.53	11671.4
	m=9	0.000552255	0.000552243	-9646.53	12640.4
	m=10	0.000552255	0.000552255	-9646.53	13581.2
$\nu = 1.8$	m=1	-2.88838	-5.93817	-10.1597	-15.5546
	⋮	⋮	⋮	⋮	⋮
	m=4	0.00145573	0.00156863	2781.07	181.002
	m=5	0.00136517	0.00136384	-8949.79	199.406
	m=6	0.00136623	0.00136626	-8949.79	219.162
	m=7	0.00136623	0.00136623	-8949.79	240.913

Table 4-5: convergence s by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 5**($\nu = 1.2, 1.5, 1.8$)

ν		N=2	N=4	N=8
$\nu = 1.2$	n=1	-0.175002	0.0168045	7.96134E-05
	n=2	-0.0590332	0.00225399	1.38225E-05
	n=4	-0.0102797	0.000260196	1.44E-06
	n=8	0.00135291	2.85067E-05	1.46E-07
	n=16	0.00264048	3.1743E-06	1.44E-07
$\nu = 1.5$	n=1	-0.329414	0.0259312	0.00016712
	n=2	-0.105287	0.00389793	2.09201E-05
	n=4	-0.0315429	0.000552252	2.77E-06
	n=8	-0.00821615	7.18969E-05	3.55E-07
	n=16	-0.00135111	8.97807E-06	4.53E-08
$\nu = 1.8$	n=1	-0.579867	0.0369195	0.000356963
	n=2	-0.180697	0.00728827	0.00005723
	n=4	-0.0647063	0.00136623	8.87E-06
	n=8	-0.0250059	0.000241561	1.37E-06
	n=16	-0.00992971	4.04809E-05	2.12E-07

Table 4-6: Maximum errors by using M-SJC of Legendre polynomials($\alpha = 0$ and $\beta = 0$) for **Example 5**($\nu = 1.2, 1.5, 1.8$)

ν		N=2	N=4	N=8
$\nu = 1.2$	n=1	0.030083	0.00266599	1.54475E-05
	n=2	0.0279257	0.000296297	1.34E-06
	n=4	0.00392711	4.21663E-05	3.20E-07
	n=8	0.00119956	6.73E-06	4.81E-08
	n=16	0.00118873	8.93E-07	1.05E-08
$\nu = 1.5$	n=1	0.0647222	0.00370846	1.05636E-05
	n=2	0.0497907	0.000296051	3.21E-06
	n=4	0.0140836	4.27645E-05	4.78E-07
	n=8	0.00357069	9.66E-06	6.27E-08
	n=16	0.000615784	1.65E-06	8.61E-09
$\nu = 1.8$	n=1	0.1244	0.00697053	2.12789E-05
	n=2	0.0840491	0.000817379	2.07E-06
	n=4	0.0312641	8.56517E-05	5.39E-07
	n=8	0.0122809	7.94E-06	1.01E-07
	n=16	0.00495367	7.62E-07	2.14E-08

V

Conclusion

In this work, we developed a new numerical method based on the shifted jacobi polynomials for solving linear and nonlinear initial value problem and boundary value problem of fractional differential equation. We extend the conventional spectral approaches such as the Shifted Jacobi Tau method for linear problem and the Shifted Jacobi Collocation method for nonlinear problem, by using the multistage methodology. In other words, on the equally spaced partition $P : 0 = x_0 < x_1 < \cdots < x_n = 1$, the convential spectral methods are applied in each subdomain, which are called the Multistage Shifted Jacobi Tau(M-SJT) and the Multistage Shifted Jacobi Collocation(M-SJC) method, respectively. From the several illustrative examples it is concluded that the M-SJT method has a better accuracy than the standard SJT for linear initial value problem and the M-SJC method gives a more accurate approximation than the standard SJC method for nonlinear initial value problem. Especially, for the less regulatity of the solution such as **Example 4**, the standard method gives a poor approximation because of smooth polynomial basis functions, whereas the proposed method obtains more accurate approximation because the polynomial basis functions can be applied on the locally smooth sub-doamin.

In addition, we extend the proposed methods for solving nonlinear boundary value problem of the fractional differential equations. Since all proposed methods are developed for solving the linitial problem, it is necessary to convert the boundary problem to the initial problem. Here we adopt the nonlinear shooting method combined with M-SJC. From the numerical example, it is clear that M-SJC has a better accuracy than standard SJC for the nonlinear boundary value problem.

It is easy to see that the matrix system by the standard SJT gets larger if many jacobi polynomials are applied as the basis functions. For the large matrix system, it usually requires a lot of computational cost. It gets worse in the nonlinear problems. In the multistage approach, however, the size of matrix system can be reduced because the method is applied to the problem only in small sub-interval. With several iteration, it requires only a simple computation. Moreover, the proposed method enhance the computatonal accuracy.

References

[1] E.H. Doha, A.H. Bhrawy, D.baleanu, S.S. Ezz-Eldien, *On shifted Jacobi spectral approximations for solving fractional differential equations*, *Comput. Math. Appl.* **219**,(2013) 8042 – 8056.

[2] A.H. Bhrawy, A.S. Alofi, S.S. Ezz-Eldien, *A quadrature tau method for variable coefficients fractional differential equations*, *Appl. Math. Lett.* **24**,(2011) 2146-2152.

[3] E.H. Doha, A.H. Bhrawy, S.S. Ezz-Eldien, *A Chebyshev spectral method based on operational matrix for initial and boundary value problems of fractional order*, *Comput. Math. Appl.* **62**,(2011) 2364-2373.

[4] Kai Diethelm, *The Analysis of Fractional Differential Equations*, *Springer,* **2010**. 3, 4

[5] Bongsoo Jang, *Efficient analytic method for solving nonlinear fractional differential equations*, *Appl. Math. Model.* **38**(2014)1775–1787.

[6] P.K. Gupta, M. Singh, *Homotopy perturbation method for fractional FornbergWhitham equation*, *Comput. Math. Appl.* **61**(2011)250–254. 1

[7] Z. Odibat, S. Momani, *Modified homotopy perturbation method: application to quadratic Riccati differential equation of fractional order*, *Chaos Solitons Fract.* **36**, (2008) 167-174. 1

[8] N.T. Shawagfeh, *Analytical approximate solutions for nonlinear fractional differential equations*, *Appl. Math. Comput.* **131**,(2002) 517-529.. 1

[9] Y. Khan, J. Diblik, N. Faraz, Z. Smarda, *An efficient new perturbative Laplace method for spacetime fractional telegraph equations*, *Adv. Difference Equ.* **2012**, (2012) 204.

[10] S. Yanga, A. Xiao, H. Su,*Convergence of the variational iteration method for solving multi-order fractional differential equations*, *Comput. Math. Appl.* **60**, (2010) 2871-2879. 1

[11] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, *Spectral Methods in Fluid Dynamics*, *Springer, New York,* **1988**. 1

[12] H.M. Jaradat, Fadi Awawdeh, E.A. Rawashdeh, *An analytical scheme for multi-order fractional differential equations*, *Tamsui Oxford J. Math. Sci.* **26**,(2010) 305-320. 1

[13] A.H. Bhrawy, A.S. Alofi, *A JacobiGauss collocation method for solving nonlinear LaneEmden type equations*, *Commun. Nonlinear Sci. Numer. Simul.* **17**(2012)62–70.

[14] Abbas Saadatmandi, Mehdi Dehghan, *A new operational matrix for solving fractional-order differential equations*, *Comput. Math. Appl.* **59**(2010)1326–1336.

[15] Salih Yalcinbas, Mehmet Sezer, Huseyin Hilmi Sorkun, *Legendre polynomial solutions of high-order linear Fredholm integro-differential equations*, *Appl. Math. Comput.* **210**(2009)334–349.

ACKNOWLEDGEMENTS

First of all, I would like to express my greatest appreciation to my advisor Prof. Boongsoo Jang for the his continuous support and enthusiasm. His guidance helped me in all the time of research. This thesis was not possible without his feedback. Also, I wish to show my sincere thanks to my committee members, Prof. Pilwon Kim and Prof. Byoung Ki Seo for their encouragement, insightful comments and constructive questions.

In addition, I thank all of lab members, who helped me a lot to prepare my thesis and defense presentation. I specially thank Kyunghoon Kim, Hyojung Lee and Thien Binh Nguyen for their encouragement and comments over the period of my master course. In addition, my appreciation goes to my friends(M_GO) for the pleasant time together.

Finally, I wish to thank to my family.